

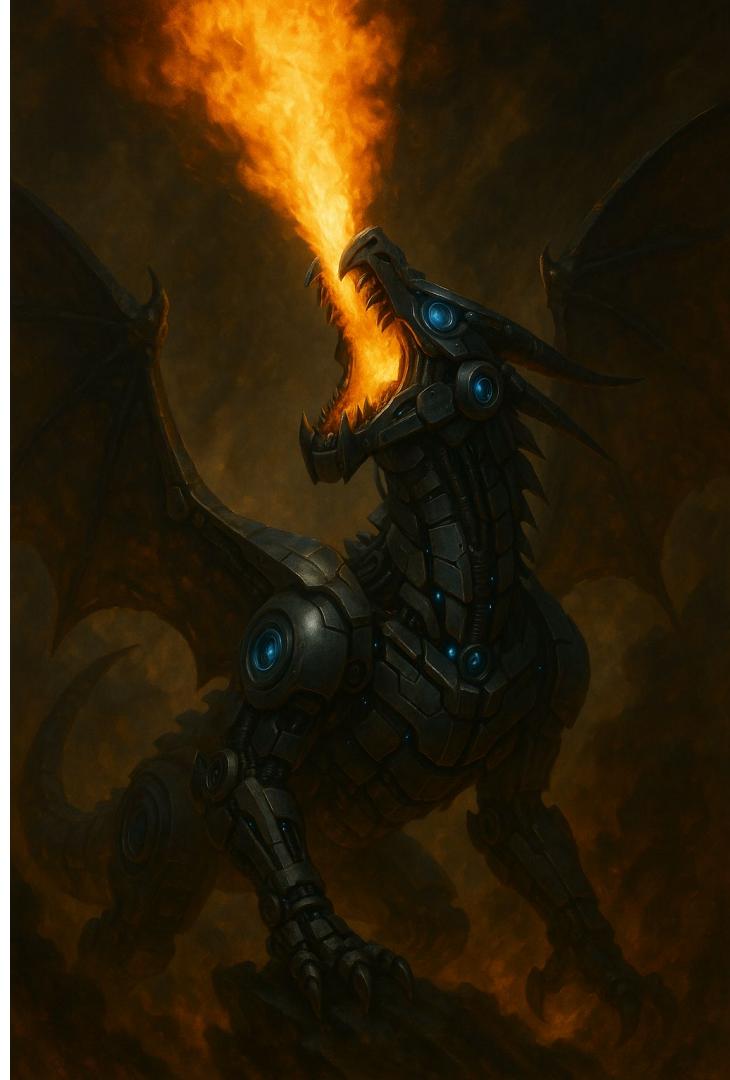


Software Craftsmanship for Coding Agents: Taming the Dragon

Steven Diamante

Technical Coach & AI Consultant

Founder of Diamante Technical Coaching



This talk is for software engineers

Thinking about using coding agents

Using agents for side projects only

Using agents in **production**, but struggling

Using agents successfully, want to hear how others do it

Who am I?

- ❑ **Steven Diamante from Raleigh, NC**
 - ❑ Founder of Diamante Technical Coaching
 - ❑ AI Consultant at Artium AI
- ❑ **Avid XP practitioner**
 - ❑ pairing
 - ❑ TDD
 - ❑ refactoring
 - ❑ continuous integration
- ❑ **Over 1k hours of Claude Code usage**
 - ❑ Building AI with AI in production
- ❑ **Hobbies**
 - ❑ Plays guitar and sings
 - ❑ Meditating for over 5 years
 - ❑ Chess and Poker player
 - ❑ Travels the world with family



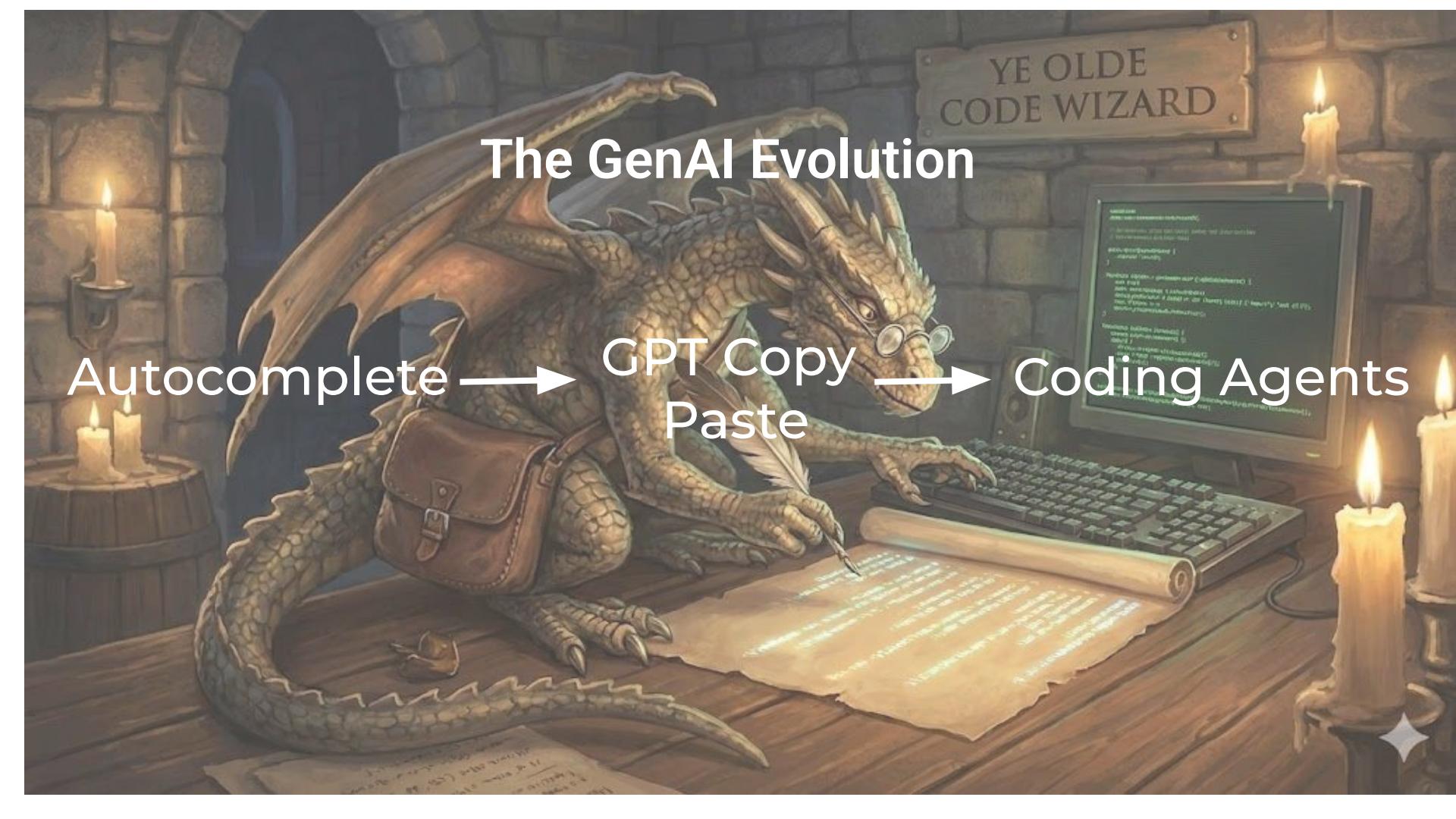
Agenda

- ❑ What are coding agents?
- ❑ My AI Coding Journey
- ❑ Common Failure Modes
- ❑ Old Habits vs. New Habits
- ❑ What is Software Craftsmanship?
- ❑ Demo: Add a new feature to a Stock Tracker App
- ❑ Taming the Agentic Dragon Workflow

Get the
Resources
& Let's
Connect



- [!\[\]\(86b7331e04fe40a56bcff2e9c065738b_img.jpg\) Connect on LinkedIn](#)
- [!\[\]\(92f87f30b7499b35d0173f4346c498d6_img.jpg\) Diamante Technical Coaching](#)
- [!\[\]\(497b6684f704c0aa6fbea9f0fd4d56c7_img.jpg\) Slides/Code from this talk](#)
- [!\[\]\(4320279ad715106747262028f44bd102_img.jpg\) Claude Code Skills & Agents](#)
- [!\[\]\(25e9c4c673069177325c65bf4771169e_img.jpg\) AI Coding Newsletter](#)
- [!\[\]\(6b6b004b0c53329d45621b2f7dfbf9f0_img.jpg\) Book a session](#)



YE OLDE
CODE WIZARD

The GenAI Evolution

Autocomplete → GPT Copy Paste → Coding Agents

Coding Agents are here!

An agent runs tools in a loop to achieve a goal.
Coding agents can read,
modify, explore your
codebase.



How can we partner with
coding agents to deliver
high-quality production code
at a sustainable pace?



Caveats

90% of production code in TypeScript, 10% in Python

Green field Microservices / micro frontends

Using Claude Code,
but techniques apply to other coding agents

Just sharing, find your own way

My AI Coding Journey

2018: Started practicing Extreme Programming

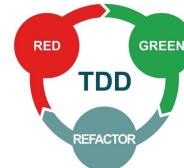
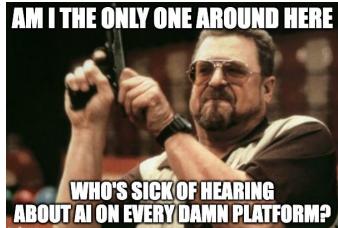
2023: Coaching software teams on technical practices



ChatGPT



Dec 2022	Feb 2025	Mar 2025	Apr 2025	May 2025	Sep 2025	Oct 2025
First used ChatGPT <i>"Didn't know what to ask it"</i>	"Sick of hearing about AI" Claude Code released	Saw coding agent demo Agile Open Northwest	Started AI consulting Built RAG pipeline in 1 week	Installed Claude Code Started custom workflows	TDD + agents workflow Parallel agents, small batches	40+ features shipped One month



Common AI Coding **Failure Modes** for Software Engineers



One shot attempts



› Create a complete flappy bird clone. Don't stop until you're done. █

```
~/p/flappy-bird-clone | Model: Sonnet 4.5 | Ctx: 0.0%
(no git) | Cost: $0.00 | v2.1.5
└ no git | Style: default | Ctx: 0
►▶ bypass permissions on (shift+tab to cycle)
```

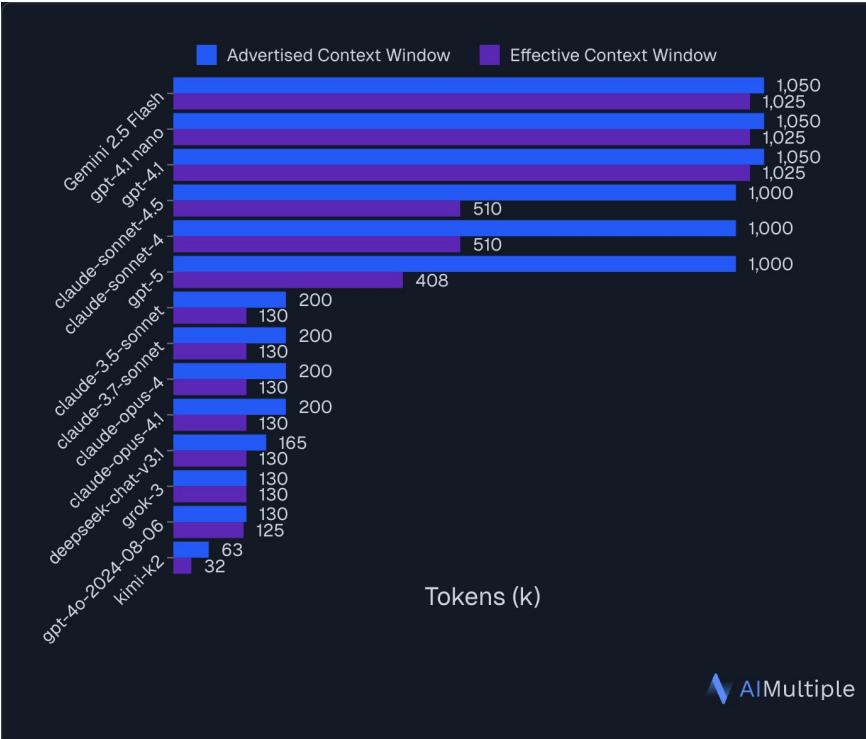
Failure Modes

Not respecting the context window



Failure Modes

Not respecting the context window



“Most models break much earlier than advertised. A model claiming 200k tokens typically becomes unreliable around **130k**, with **sudden performance drops** rather than gradual degradation.”

— AIMultiple

Failure Modes

No ground rules



AGENTS.md

CLAUDE.md

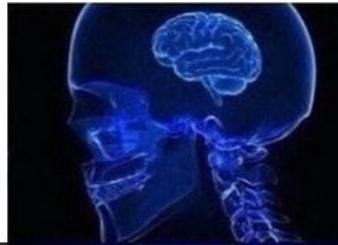
.windsurfrules

Failure Modes

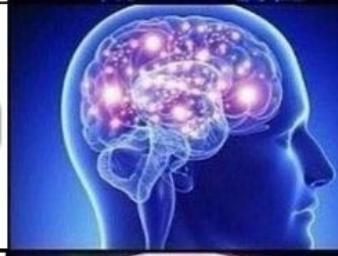
Telling the agent what to code

Failure Modes

**WRITE
THIS EXACT
FUNCTION FOR ME**



**IMPLEMENT
AUTHENTICATION
USING JWT**



**HERE'S THE
PROBLEM: USERS
NEED SECURE LOGIN**



**INTERVIEW
ME ABOUT
REQUIREMENTS,
PROPOSE 3 SOLUTIONS**



Old Habits vs. New Habits



Task Management

OLD HABIT

- ❑ Task lists
- ❑ Test lists
- ❑ Behavior-first thinking



NEW HABIT

- ❑ Task lists turn into .md files with ASCII wireframes
- ❑ Behaviors and outcomes thinking still remains

Daily Pairing

OLD HABIT

7+ years of pair programming

- Communication
- Critique
- Honest feedback



NEW HABIT

- Strong-style pairing with agents
- Communicate intent, not implementation details
- Tap into agent's optionality

Design Evolution

OLD HABIT

- ❑ Evolutionary design
- ❑ Gradual refinement
- ❑ Careful iterations
- ❑ Merciless refactoring



NEW HABIT

- ❑ Incremental development with agent collaboration
- ❑ Rapid iteration cycles with instant feedback
- ❑ Seek advice of agent's vast wealth of knowledge

Commit Discipline

OLD HABIT

- Sometimes microcommits
- Occasional rollbacks
- Less disciplined



NEW HABIT

- Commit once the code works
- Must be able to rollback changes quickly
- Agents create messes fast!

Learning & Experimentation

OLD HABIT

- Careful experimentation
- Code attachment
- Cautious iterations



NEW HABIT

- Rapid experimentation
- Throw away code easily
- Keep learnings, ditch implementation

Manifesto for Software Craftsmanship

Raising the bar.

As aspiring Software Craftsmen we are raising the bar of professional software development by practicing it and helping others learn the craft. Through this work we have come to value:

Not only working software,
but also **well-crafted software**

Not only responding to change,
but also **steadily adding value**

Not only individuals and interactions,
but also **a community of professionals**

Not only customer collaboration,
but also **productive partnerships**

That is, in pursuit of the items on the left we have found the items on the right to be indispensable.

The Craftsmanship Comeback

- ❑ A movement to prioritize
 - ❑ quality
 - ❑ community
 - ❑ incremental business value
- ❑ XP practices like
 - ❑ TDD
 - ❑ Continuous Integration (TBD)
 - ❑ Refactoring
 - ❑ Social programming

Stock Tracker App

WHAT IT IS

- ❑ Modern web platform for real-time stock tracking
- ❑ Mobile-optimized for retail investors & traders
- ❑ Unified view of essential stock data

CORE CAPABILITIES

- ❑ Search & Discover Stocks
- ❑ Watchlist Management
- ❑ View interactive candlestick charts
- ❑ Comprehensive Metrics

Diamante Investments v1.0 -> v2.0

The screenshot shows a Java Swing application window titled "Diamante Investment's Stock Portfolio Portal". At the top right is a user icon labeled "Userna... sdiamante". On the left, there's a "Watch ..." panel showing stock prices for AAPL, MSFT, and NFLX. In the center, a circular callout highlights the "Summary" tab of the Apple Inc. stock details panel. The panel displays financial data including Open (180.07), Close (179.80), Day's Range (179.04 - 182.34), 52 Week Range (124.17 - 198.23), Volume (58.08M), and various moving averages. A red circle highlights the "AAPL" label at the top of the summary panel. Below the main panel, the text "StockSymbol.java" is visible.

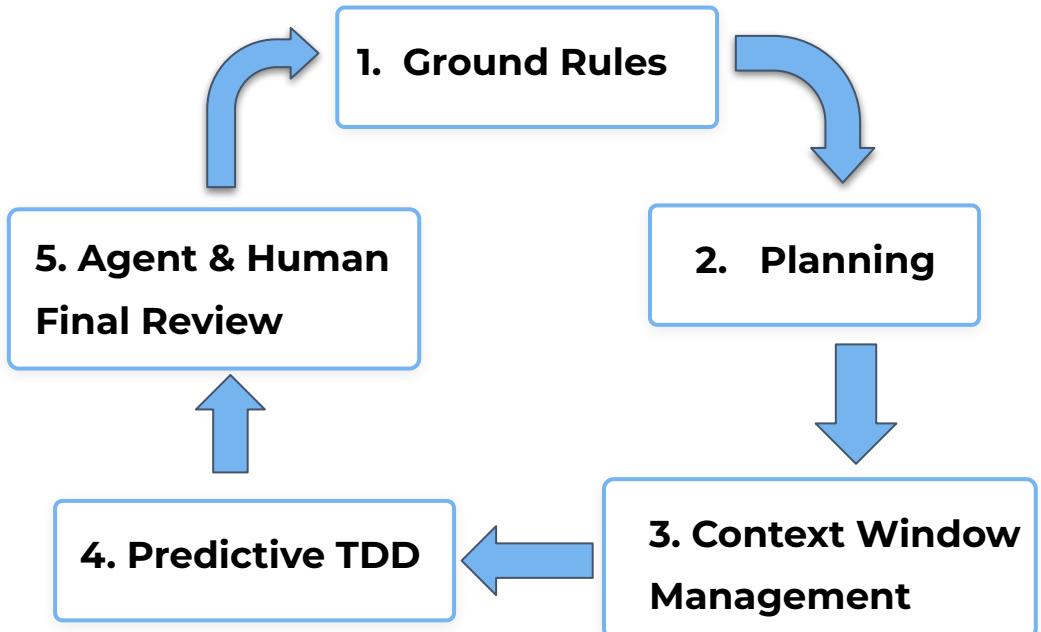
Java Swing Desktop Application built in 2017

Diamante Investments

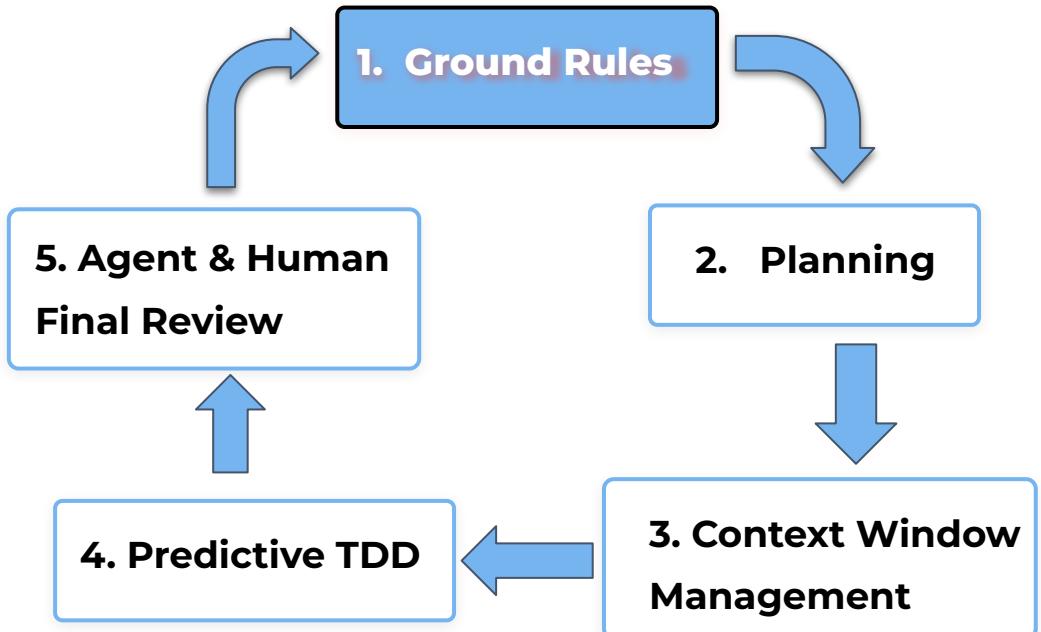
The screenshot shows a modern web-based interface for Diamante Investments. It features a search bar with "AAPL" typed in. Below it is a card for Apple Inc. (AAPL) with the following information: Price (\$283.1), Company (APPLE INC), and a change of ▲ +\$4.25 (▲ +1.52%).

Current Web version

Taming the Agentic Dragon



Taming the Agentic Dragon



Ground Rules

Don't repeat yourself in every chat

Give your agent rules and guidelines
to follow

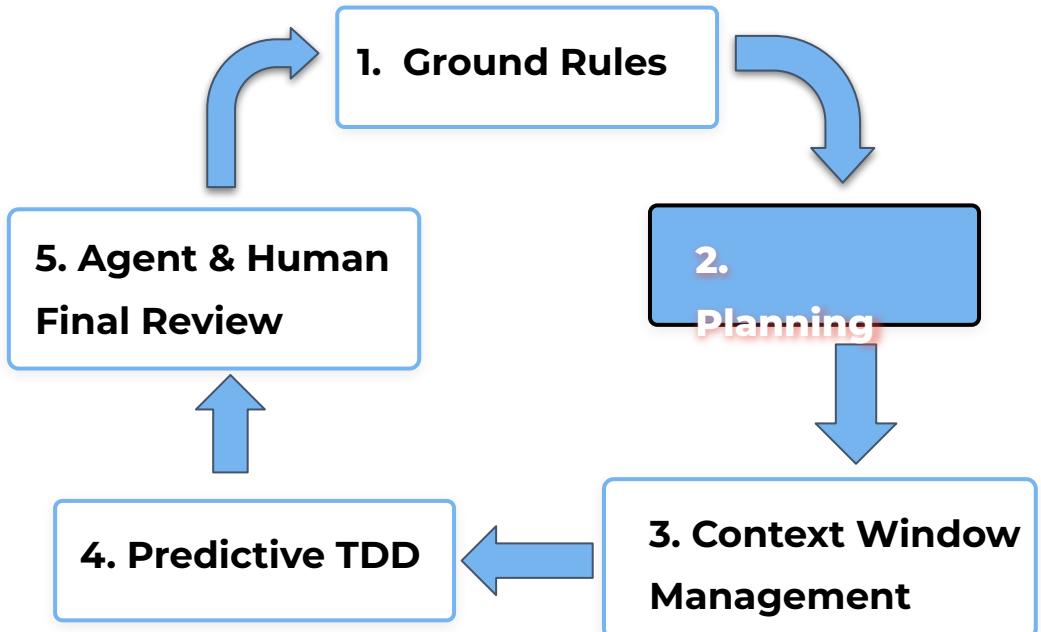
- CLAUDE.md, AGENTS.md files
- Working agreement** with agent

Ask agent to help create rules

When working with an agent's defaults, they may not suit your needs. As you find yourself teaching and lecturing the agent, make a note to put down rules.



Taming the Agentic Dragon



Define Success Before You Code



- Gherkin User Stories**
- Given/When/Then scenarios**
- Clear acceptance criteria



Expected Outcomes

What **success** looks like
Measurable results

Provide Context



Why are we doing this?

What **problem** does it solve?

Business value alignment



Integration Information

- API documentation
- System diagrams
- Existing code patterns

Interview Mode

 Thariq ✅ @trq212 · Dec 28, 2025 ⚙ ...
my favorite way to use Claude Code to build large features is spec based

start with a minimal spec or prompt and ask Claude to interview you using the AskUserQuestionTool

then make a new session to execute the spec

read this @SPEC.md and interview me in detail using the AskUserQuestionTool about literally anything: technical implementation, UI & UX, concerns, tradeoffs, etc. but make sure the questions are not obvious

be very in-depth and continue interviewing me continually until it's complete, then write the spec to the file

283 691 6.7K 1.7M ↗ ↑
[Show more replies](#)



Ask Clarifying Questions

Let agent probe for details

Uncover edge cases



Give Me 3 Options

Different **implementation** approaches

Trade-off analysis



For UI: Ask for ASCII Wireframes

Visual layout planning before implementation

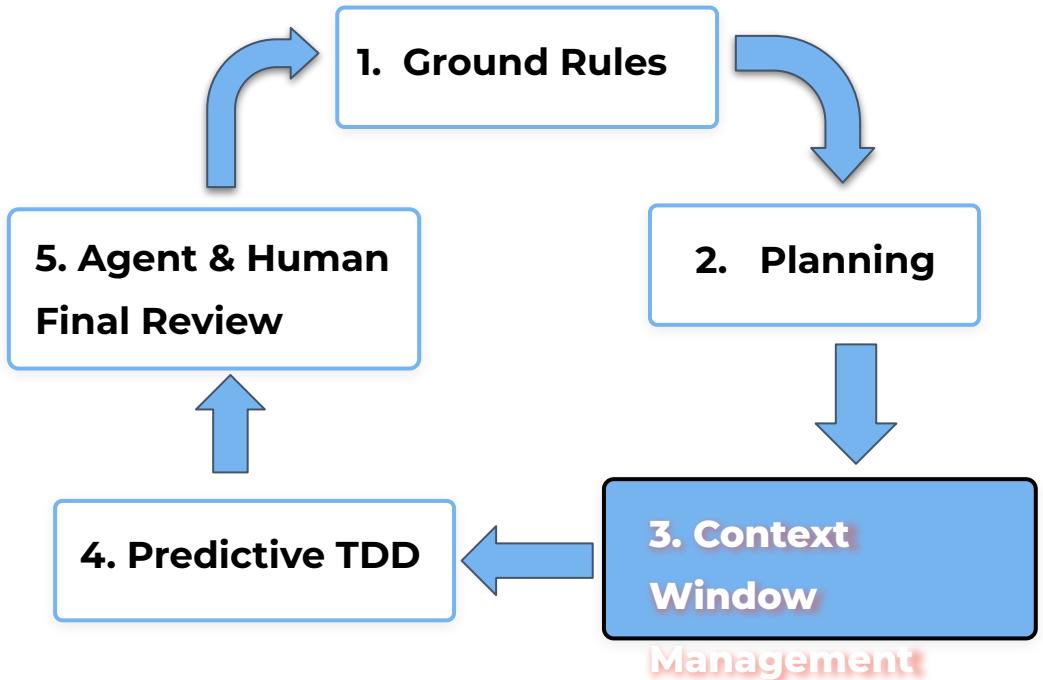
Review & Refine



WARNING: Read Every Line

-  There's something **wrong** in the plan
-  Find it and correct it
-  Challenge **assumptions**
-  Verify completeness
-  Have another agent review the plan

Taming the Agentic Dragon



Context Window Management

Plan ready? Check your context window!

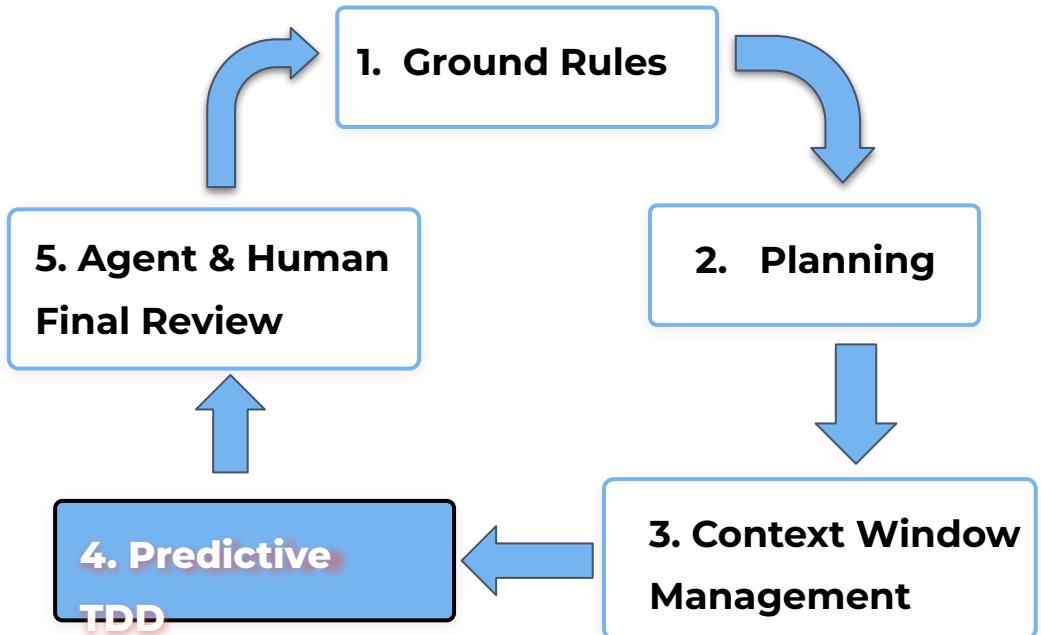
Models degrade at ~60% of context window

**Extract plan to
.md file?**

OR

**Continue with
execution in same
session?**

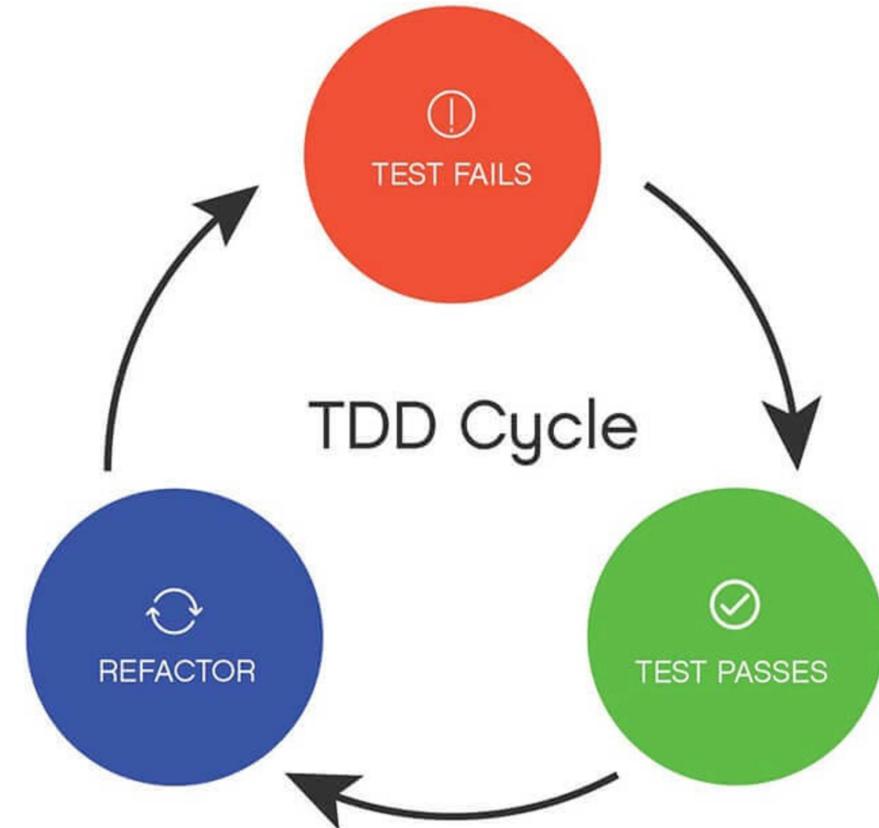
Taming the Agentic Dragon



Predictive TDD

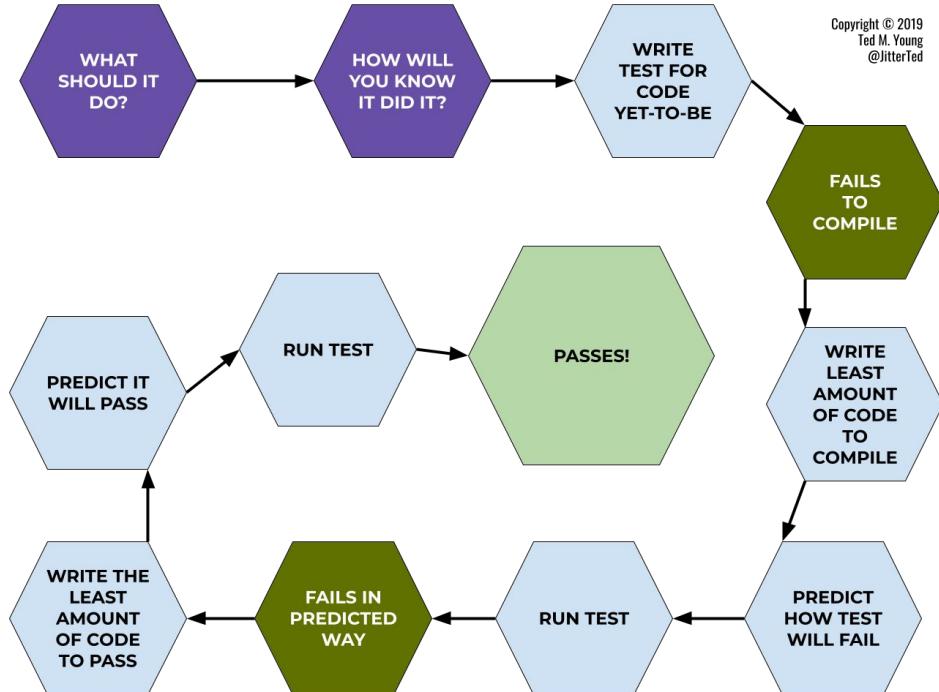
Ted M. Young (@JitterTed)

- ❑ Use custom TDD slash command
- ❑ Test **behaviors**, not implementation
- ❑ Keep mocking minimal
- ❑ Prefer acceptance tests with test containers & mock servers
- ❑ Tight feedback loop for agent to self-correct

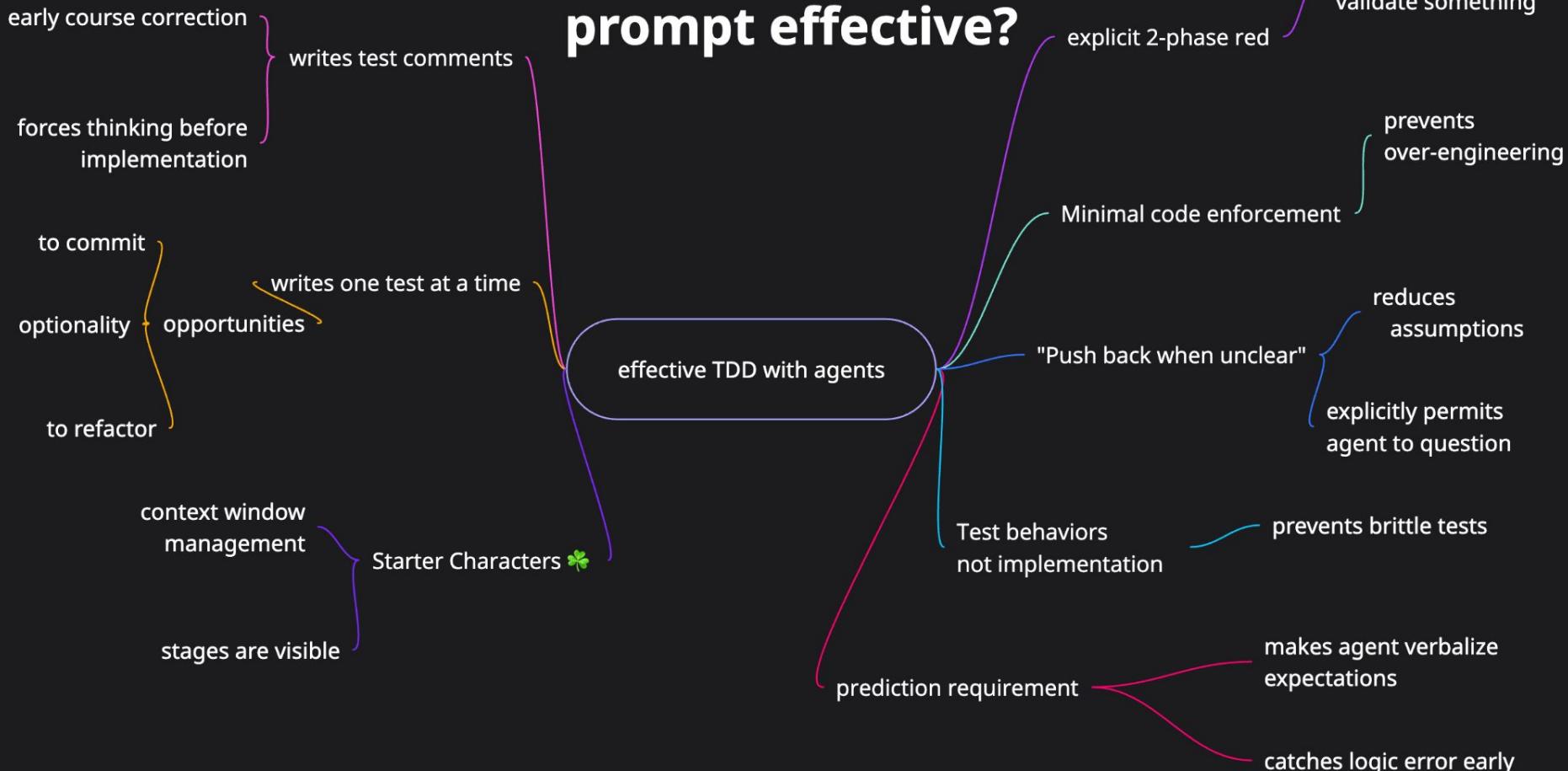


Predictive TDD

Ted M. Young (@JitterTed)



What makes this prompt effective?



How to Enforce Refactoring Step

Git hooks + strict linting rules

- ❑ precommit => build & format & lint
- ❑ prepush => run all tests

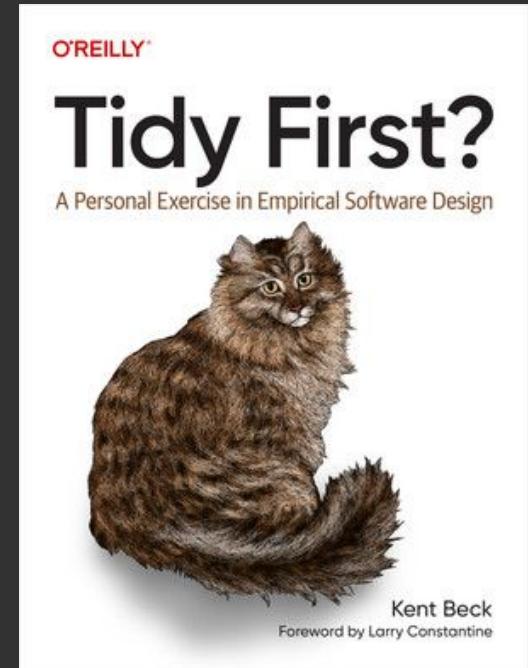


💡 Commit with additional coding agent
to save unnecessary context

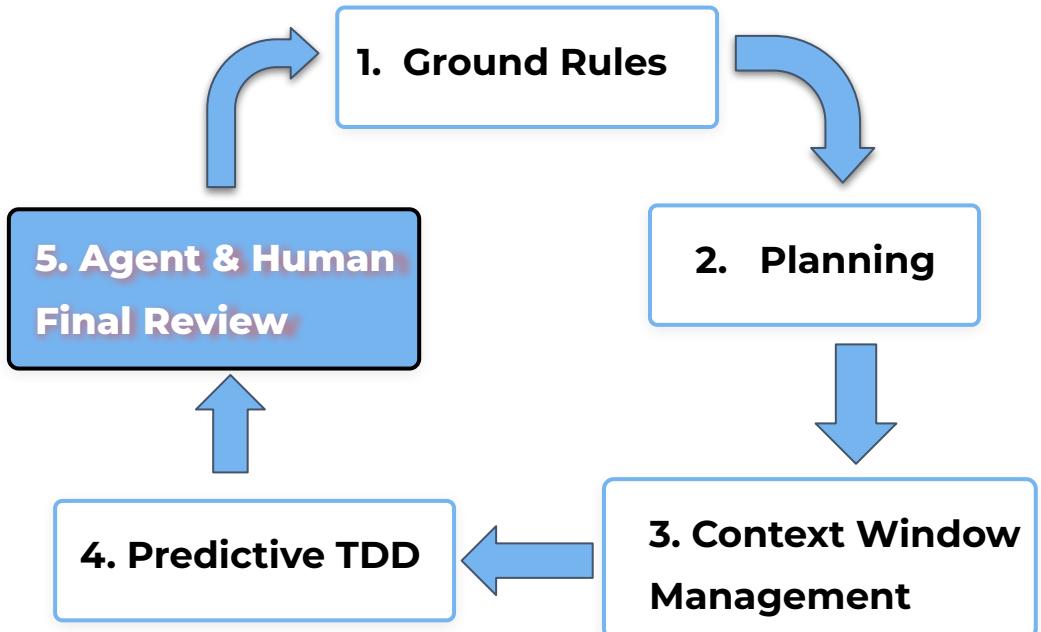
How to Enforce Refactoring Step

Tidy Up Skill

- ❑ Micro refactors covered by tests
- ❑ Microcommits using Arlo's Commit Notation (e.g. - r <refactoring>)
- ❑ If tests fail, revert
- ❑ Remove dead code, improve names, extract methods, remove duplication, etc.



Taming the Agentic Dragon



Agent & Human Final Review



Manual Testing

Verify functionality works as expected



Playwright MCP

UI testing by agent

Automated exploratory testing



Fresh Agent Review

Code review with **criteria**

Unbiased perspective

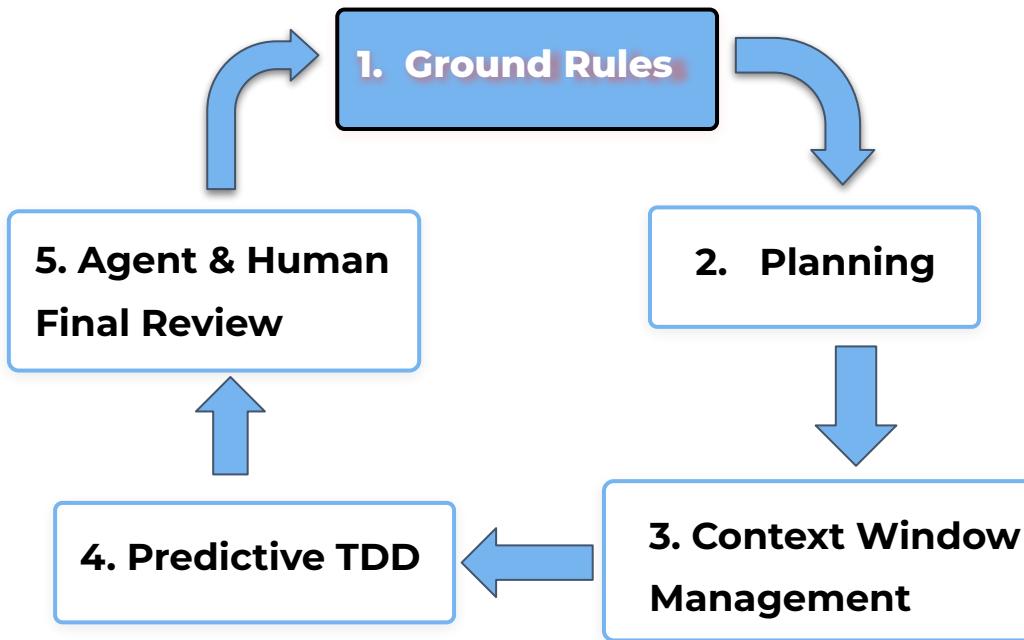


Human Final Review

Human oversight

Refactor as needed

Taming the Agentic Dragon



Remember

“I'm about to clear this conversation. Compile a list of the 5 most important things that would be useful to add to the CLAUDE.md. Tell me first for approval”

How can we partner with
coding agents to deliver
high-quality production code
at a sustainable pace?



Takeaways

1. Create ground rules, stop repeating yourself
2. Plan first: Use interview mode, get 3 options, verify with ASCII wireframes
3. Watch your context window - extract plans at ~60%
4. Use Predictive TDD to catch errors early and enable self-correction
5. Multi-layer review: manual testing → agent review → human oversight
6. Experiment and find what works for you



Tame your Dragon

Tame the Dragon

Get the
Resources
& Let's
Connect



- Connect on LinkedIn
-  Diamante Technical Coaching
-  Slides/Code from this talk
-  Claude Code Skills & Agents
-  AI Coding Newsletter
-  Book a session

Feedback?